Path Integration using Coupled Bump Attractors

NX-465 Mini-project 2

Challier Camille (311020) and Detrois Yannick (328346)

December 3, 2024

Introduction

Ants and various other animal species can navigate back to their nest after foraging trips without depending on visual cues. This ability, known as **path integration**, involves maintaining an internal estimate of the relative position of their nest while moving. It is essential for this memory mechanism to be capable of rapid updates while reliably maintaining its state over an extended period. A prevalent theory regarding the neural substrate of this memory suggests that the information is preserved in a bump attractor's state [3]. Bump attractors are recurrent neural networks that preserve a localised peak of neural activity. They can encode an agent's spatial position by representing it as the position of this 'activity bump' within the network. To verify this hypothesis, we have simulated this process by implementing a neural circuit that integrates the ant's movements in two dimensions to continuously estimate its current position.

0 Poisson neurons

We'll begin by simulating Poisson neurons to mimic neural activity. Despite some biological inaccuracies, their simplicity allows for efficient modelling of large-scale population dynamics.

0.1. Transfer function g. The transfer function g used in the expression of instantaneous mean firing rate is a function of the potential h and is defined as $g(h) = \frac{1}{1+e^{-2\alpha(h-\beta)}}$. Both parameters α and β affect the shape of the activation function and thereby on the simulation dynamics. In the following we assume $\alpha \geq 0$. For $\alpha < 0$, the effect is the same but mirrored along the $h = 0$ axis. α determines the 'slope' of the sigmoid function, a larger value leading to a more abrupt transition and vice versa (Appendix A). The function varies between a step function for $\alpha \to \infty$ to a constant function of value $\frac{1}{2}$, for $\alpha = 0$. Meanwhile, β determines the shift of the sigmoid function along the x-axis. Increasing β shifts the distribution centre to the right, while decreasing it shifts it to the left. These parameters allow better control over the behaviour of the sigmoid function, making it more or less sensitive to changes in the input, and shifting its midpoint along the x-axis.

0.2. Non-recurrent network. The theoretical rate follows the slowly oscillating input $I_i(t) = I_0 \sin(\omega t)$ (ω=0.01 rad/ms) with a small delay (Appendix B). It however follows it only when it the input is in the interval [0, 1] - when it is outside the range, in [1, 2] or [-1, 0], the firing rate stays at 1 or 0 respectively because the theoretical firing rate is restricted to [0, 1]. The simulated mean firing rates with 100 and 1000 neurons follow similar dynamics as the theory, however with considerably more noise. When at the maximum observed firing rate, the fluctuations around the theoretical value of 1 are in the order of ~ 0.3 for $N = 100$ and ~ 0.1 for $N = 1000$. Hence, statistical fluctuations become smaller for larger networks and converge to the smooth theoretical firing rate. Indeed, each neuron representing a "sample" of the firing distribution, increasing the number of samples implies converging to the distribution mean with decreasing standard deviation.

1 Bump attractor

figures/1_1.png

Figure 1: Raster plot of the dynamics of recurrent networks with $N = 300$ neurons with no external input. Three values of the interaction strength ($J = [4.5, 4.75, 5]$) are tested for bump stability. Bins of 1 ms were used for firing rate averaging.

1.1. Recurrent network. The recurrent network with no external input produces a jump for $J \geq 4.75$ pC as can be seen in Figure 1. Unless specified otherwise the value of J will be $J = 5$ pC.

1.2. Bump location. With the reference settings given in the exercise ($N = 300$ neurons, $\Delta t = 0.1$ s, $\tau = 10$ s), the location of the bump is generally stable, with only a small drift, as can be seen in the *Reference* panel of Figure 2. To quantify the stability we used the mean median absolute deviation (MAD), a measure of the variability of a univariate sample of quantitative data used in Figure 2 of [3]. We found a mean MAD = 0.0993 ± 0.0534, which is very small compared to the scale of the bump location $\theta \in [0, 2\pi)$.

1.3. Bump drift. The inherent stochasticity of the system can cause small fluctuations at each step that figures/1_3.png

Figure 2: Impact of experimental parameters on bump location. The reference settings are as given in the exercise ($N = 300$ neurons, $\Delta t = 0.1$ s, $\tau = 10$ s) and a single parameter is changed per panel to asses its effect on bump stability. The four panels are, from left to right: reference values, $\Delta t \rightarrow 0.01$ s , $\tau \rightarrow 1$ s and $N \rightarrow 3000$ neurons. For the raster plots, bins of 1 ms were used for firing rate averaging while for the bump location the bins were of 10 ms.

have, over several time steps, an additive effect that causes the observed drift in the bump location. Many experimental parameters have an effect on bump stability, as pictured in Figure 2. Generally, parameter modifications that reduce the impact of noise in the system (such as increasing the number of neurons N) also improve the stability of the bump, as can be observed in the right-most panel of Figure 2.

All of the parameters have an effect on bump stability. Reducing the time constant τ means that the system more quickly changes its potentials $h_i(t)$, which can be seen in Figure 2 through a faster changing (or more unstable) bump. Increasing the number of neurons N stabilises the bump location because an increased number of neurons reduces the effect of stochasticity. Indeed, the bump location is much more stable using 10 times more neurons than reference in Figure 2. The time step Δt does not have such a clear effect as the other parameters, because the stochasticity at each step depends on Δt through $\mathrm{P}\{\text{spike in } [t, t + \Delta t)\} = r(t) \cdot \Delta t$. This means that the *average* random drift will be the same over a same time interval. Stability is therefore increased with increasing the number of neurons N and with increasing the system's time constant τ, amongst others.

figures/1_4.png

Figure 3: Spike activity and connectivity profile of a network with external input. The first (red) external input is a normal distribution with mean $\mu_1 = \frac{2\pi}{3}$ and $\sigma = \frac{\pi}{8}$ applied between 300 and 400 ms. The second (orange) is a normal distribution with $\mu_2 = \frac{4\pi}{3}$ and $\sigma = \frac{\pi}{8}$ applied between 600 and 700 ms.

1.4. Recurrent network with external current. The external input can change the bump location as its magnitude depends on the neurons' position (connectivity profile in Figure 3). This implies that neurons near the distribution centres μ_1, μ_2 will be stimulated more than by the external input neurons far away. This explains why the bumps appear near the distribution centres.

figures/1_5.png

Figure 4: Effect of small shift in connectivity on spiking dynamics. The leftmost panel represents the baseline with no shift, the following from left to right have shifts $\varphi = [\frac{\pi}{64}, \frac{\pi}{32}, \frac{\pi}{16}]$.

1.5. Connectivity with phase difference. The model with phase difference can be obtained by replacing $\cos(x_i) \rightarrow \cos(x_i - \varphi)$ and $\sin(x_i) \rightarrow \sin(x_i - \varphi)$ in the expression of the recurrent input. For a small angle φ, we expect a neuron to be more strongly affected by neurons that are not anymore its direct neighbours, but distributed around a maximum shifted by φ. Usually, a neuron n_i is most strongly affected by neurons n_j that are closest to it (i.e. n_j such that $(x_i - x_j) \mod (2\pi) \approx 0 \iff \cos(x_i - x_j) \approx 1$). The shift means that instead, n_i will rather be more strongly affected by for instance neuron n_{j+5} than neuron n_j, which over time creates unstable wave-like patterns (Figure 4). The bump indeed propagates in the neuron population because the firing activity propagates from e.g. neuron $n_{j+5} \rightarrow n_j \rightarrow n_{j-5} \rightarrow \ldots$

1.6. Line model. For the Gaussian turning curve, the bumps are stable for instance for the values $J_0 = -1$, $J_1 = 3$ and $\sigma = 1$ as shown in Appendix C. For the line model there is no connectivity between neurons located at around 2π and 0 as in the ring model, meaning such neurons no longer positively interact. A difference that can thus be observed is the the bump location is mostly around the centre (where connectivity is strongly positive), while in the ring model the bump could also occur at the borders (Appendix C).

2 Integration

Unless specified otherwise the value used in this exercises will be T = 1000 ms, Δt=0.1 ms, N = 300 neurons, $\theta = 10°$ and J = 3 pC.

2.1. Push-pull system. The weights affecting neurons in the left and right population are $w_{L\to L}, w_{R\to L}$ and $w_{R\to R}, w_{L\to R}$ respectively, all defined as:

$$\begin{aligned} w_{A\to L}(x_i^L, x_j^A) &= \cos(x_i^L + \theta - x_j^A) \\ &= \cos(x_i^L + \theta)\cos(x_j^A) + \sin(x_i^L + \theta)\sin(x_j^A) \\ w_{A\to R}(x_i^R, x_j^A) &= \cos(x_i^R - \theta - x_j^A) \\ &= \cos(x_i^R - \theta)\cos(x_j^A) + \sin(x_i^R - \theta)\sin(x_j^A) \end{aligned}$$

where $A \in \{L, R\}$. The inputs of the left ($A = L, \bar{A} = R$ and using + sign) and right ($A = R, \bar{A} = L$ and using − sign) population are given by the following expressions. For simplicity we ignore the external stimulation $I_{\text{ext},i}^A(t)$ for both populations and abbreviate the notation using $w_{A\to B}(x_i^B, x_j^A) = w_{A\to B}$.

$$\begin{aligned} I_i^A(t) &= \frac{J}{N}\sum_{j=1}^{N}\Big[w_{A\to A}S_j^A(t) + w_{\bar{A}\to A}S_j^{\bar{A}}(t)\Big] \\ &= J\big(\cos(x_i^A \pm \theta)[m_{\cos}^L(t) + m_{\cos}^R(t)] \\ &\quad + \sin(x_i^A \pm \theta)[m_{\sin}^L(t) + m_{\sin}^R(t)]\big) \end{aligned}$$

where $m_f^A(t) = \frac{1}{N}\sum_{j=1}^{N} f(x_j^A)S_j^A(t)$ with $A \in \{L, R\}$ and $f(x) = \{\sin(x), \cos(x)\}$.

Weights reaching the L population have the maximum connectivity when the difference between the connected neurons is $-\theta$. For the R population in turn, the maximum is for a difference of θ. The equilibrium to stably maintain this shift comes from the interaction term between the populations, which is where the **push/pull mechanics** come in place. Indeed, when one population (say, R) is completely inhibited, only the term $w_{L\to L}$ is left for population L, which corresponds to the case with phase difference described under section 1.5 and is not stable. Similarly, if population L is inhibited, population R will shift but in the opposite direction due to the opposing sign. Hence, populations naturally "pull" apart but are "pushed" together by the interactions in between populations to have a stable overall network. An only *partial* inhibition of one population therefore results in a small shift towards that population, which is a useful property to integrate said shift. Note that the same logic applies in the opposite directions for excitation instead of inhibition. Hence, when one population is excited, or "pushed" by an external input, the other population is simultaneously inhibited/"pulled" [1].

2.2. Combined circuit stability. As depicted in Figure 5, the mean MAD of the bump location for 100 runs is of 0.052 radians for both the left and right population. The typical deviation of the bumps' locations from the median is very small, indicating that the bumps' positions are stable over simulation runs. Note that the mean MAD is roughly halved compared to the single neuron setup, showing that the combined circuit has indeed increased stability.

figures/2_2.png

Figure 5: Stability Analysis of two offset bump attractors network with randomly initialised potentials. The first panel illustrates an example of the bump location over time for a single run, while the second panel displays the mean median absolute deviation of the bump location across 100 runs.

2.3. Bump initialisation. To initialise bumps such that their bump location mean $\theta_{\text{bump mean}} \approx \pi$, we can set the initial voltage of each neuron to follow a scaled normal distribution $\sigma\sqrt{2\pi} \cdot \mathcal{N}(\mu = \pi, \sigma = \frac{\pi}{10})$ centred around the neuron i with position $x_i \approx \pi$. The previously analysed circuit stability allows for the bump to stably maintain the initial condition. Using this initialisation, we find over 100 standard runs a mean of the bump location mean of 180.54° ± 6.20° which is close to the expected 180°.

figures/2_4.png

Figure 6: Stability Analysis of two offset bump attractors network with randomly initialised potentials.

2.4. Bump location Simulation. Figure 6 shows that the relation between I_0 and $\theta_{\text{bump mean}}$ is approximately linear (with $R^2 = 0.93$) in the interval [-0.65 nA, 0.65 nA], meaning a good upper limit on the input strength I_0 is 0.65 nA. We can see in Figure 12 of Appendix D that the transition is stable only for the values falling in above specified interval.

2.5. System for input's integration. Offset bump attractors models can maintain a continuous range of activity states, typically visualised as localised "bumps" of neural activity. These models convert transient inputs into persistent changes in activity. The bumps can move smoothly in response to inputs, allowing the model to encode and update activity based on the input. As seen in the previous question, the bump location encode linearly the input stimulation.

Consequently, the position of each bump represents the integrated value of its respective input.

3 Path integration

The parameters used in this exercises will be T = 1000 ms, Δt=0.1 ms and N = 300 neurons, unless specified otherwise. For the head population the values are I_0= 1 nA and J = 5 pC. For the integrator, the parameter θ is set to 10°.

3.1. Smooth random 2D trajectory. To generate a numerical random trajectory that is sufficiently smooth for our agent, we used the following algorithm.

Algorithm 1 Generate Smooth Random Trajectory

```
Require: $T$, $\Delta t$, $\alpha$, $\max \Delta\theta$
  num steps ← $\frac{T}{\Delta t}$
  $\vec{p}(0)$ ← random uniform in $[0,1] \times [0,1]$
  $\theta_H(0)$ ← random uniform in $[0, 2\pi]$
  for $t = 1$ to num steps do
    $\Delta\theta$ ← random uniform $[-\max\Delta\theta, \max\Delta\theta]$
    $\theta_H(t) \leftarrow \theta_H(t-1) + \Delta\theta$
    $d\vec{p} \leftarrow \alpha \cdot \Delta t \cdot [\cos(\theta_H(t)), \sin(\theta_H(t))]$
    $\vec{p}(t) \leftarrow \vec{p}(t-1) + d\vec{p}$
  return $\theta_H$, $\vec{p}$
```

Parameter α ensures that the step size is not too big, which would result in a trajectory scale that the agent cannot integrate. To enforce smoothness, the maximum head direction variation $\max \Delta\theta$ should be small, limiting the trajectory change at each step. A step size $\alpha = 0.01$ and $\max \Delta\theta = \frac{\pi}{60}$ gave a sufficiently smooth curve without abrupt turns and will be used in the following, unless specified otherwise.

3.2. Head direction cells simulation. As shown in panel c of Figure 8, the bump location θ_{bump} in head direction cells accurately corresponds to and encodes the current head direction θ_H.

3.3. $\mathbf{J_{head}}$ Value. In section 2, we established that the input I_0 must lie within the interval [-0.65 nA, 0.65 nA] to remain in the linear regime. In Figure 7, we observed that this interval is strictly maintained for $\text{J}_{\text{head}} \leq 1.75$ pC and that most of the input lies within the interval for $\text{J}_{\text{head}} = 2$ pC. According to Figure 7 in Appendix E, the linear regime seems to be maintained up to $\text{J}_{\text{head}} = 2$ pC. We will use $\text{J}_{\text{head}} = 1.75$ pC in the following to maximise the interval of the linear regime.

3.4. Full circuit equations. The input of the head direction cells is given by :

$$I_i^{\text{head}}(t) = J\big(\cos(x_i) m_{\cos}(t) + \sin(x_i) m_{\sin}(t)\big) \\ + I_0\big(\cos(x_i - \theta_H)\big)$$

The inputs of the left ($A = L$) and right ($A = R$) population for integration of the position of the agent is given by the following expression (with the upper symbol in ± and ∓ corresponding to L).

$$I_i^A(t) = J\big(\cos(x_i^A \pm \theta)[m_{\cos}^L(t) + m_{\cos}^R(t)] \\ + \sin(x_i^A \pm \theta)[m_{\sin}^L(t) + m_{\sin}^R(t)]\big) \\ \mp \frac{J_{\text{head}}}{N} \sum_{j=1}^{N} \big[(S_j^{\text{head}}(t) f(x_j^H)\big]$$

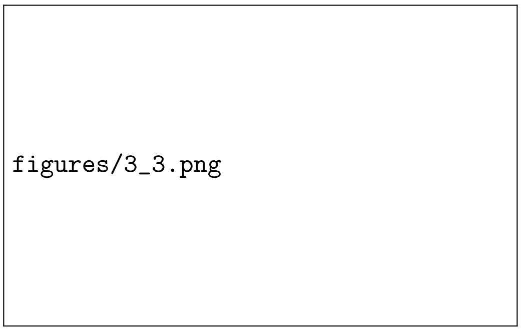

Figure 7: Determining the Optimal J_{head} value for Linear Regime Input from Head Direction Cells. The range of J_{head} tested spans from 1 to 2 pC. The dashed lines represent the interval [-0.65 nA, 0.65 nA] where the input I0 must lie to remain in the linear regime.

For the x-coordinate, function $f(x) = \cos(x)$ and for the y-coordinate, $f(x) = \sin(x)$. Furthermore, $m_f^A(t)$ is such as described in section 2.1 and $S_j(t) = \frac{1}{\Delta t}$ if there was a spike of neuron i in time step k.

3.5. Trajectory decoding. Illustrated in Figure 8, the encoded bump location closely mirrors the position of the ant, demonstrating the successful integration of its path within the model and its capability to memorise its relative position over time. Additionally, in panels d and e, a linear relationship between the ant's x-position and the bump location can be observed. While the decoded path is generally very similar to the real path, there are occasional abrupt location changes and slight shifts apparent in the trajectory. The decoded path's accuracy declines when handling complex paths, such as those with rounded trajectories.

As seen in section 1.3, increasing the number of neurons N is likely to improve the path integration capacities of the model. Similarly, increasing τ could refine the smoothness of the decoded path, and decreasing the integration time T increase the stability. Finally, changing the population offset θ could also influence the stability of the network.

3.6. Biologically realistic model? The model contains both idealised aspects and biologically plausible implementations. The head direction cell for instance is, in our case, idealised because it follows a single numerical value that perfectly represents the path. However, biological head direction cells with a similar function have been documented in insects, notably *Cataglyphis fortis* which uses polarised sunlight recep-

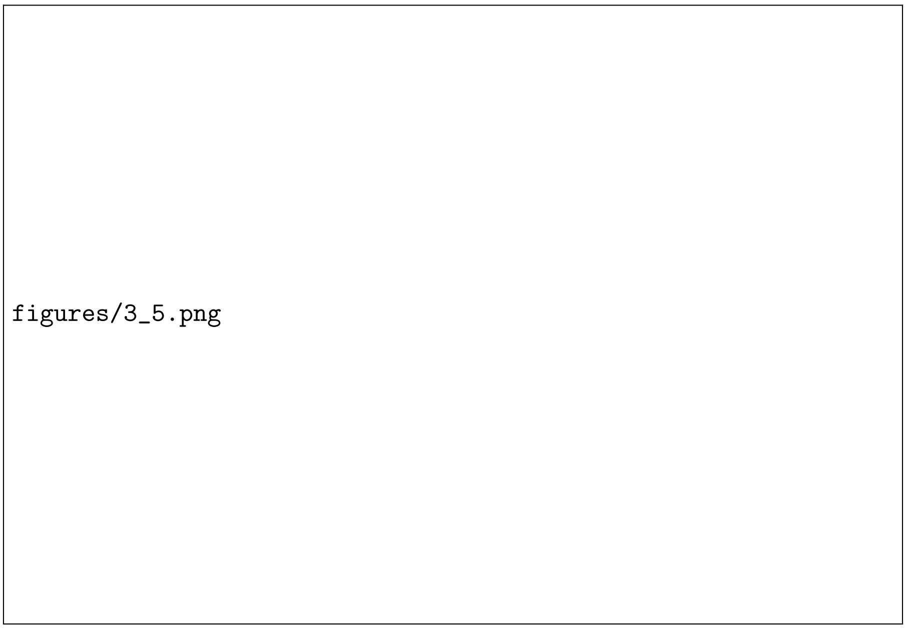

Figure 8: Simulation of the activity of the network for the whole trajectory. Parameters : N=300, Jhead = 1.75 pC. (a) Mean bump location of the right and left populations in the two positions integrators over time. (b) x and y positions of the ant over time. (c) Comparison of the Current Head Direction θ_H and the Bump Location θ_{bump} in Head Direction Cells Simulation. (d,e) Bump location as a function of the position and its linear fit. (f) Decoded and original trajectories.

tors to obtain directional information [2]. Another idealisation is the omission of external noise, which doesn't allow to test the model's robustness against it [4]. Other observed aspects, such as memory decay and path integration are also modelled in our simulations. Many other model aspects however are only ideal models of biological systems, such as the neuron connectivity, the Poisson neuron firing model, the neuron interaction dynamics, or the simultaneous updating, amongst others.

3.7. Tuning a larger network. The values $\theta = 15°$ and $\tau = 12$ were determined through hyperparameter tuning, achieved by minimizing the sum of the Euclidean distance between the decoded and initial paths. As illustrated in Appendix F, in the case of a path of 3 seconds, increasing the number of neurons N improves the accuracy of the decoded path. Longer paths are more challenging to accurately reconstruct, necessitating a larger number of neurons N. More globally, a larger N improves significantly the model's performance compared to Figure 8. The decoded path is smoother and closely follows the initial path with better precision.

Conclusion

Coupled bump attractors are able to follow and integrate a sequence of values that allow them to memorise and reconstruct a path. Several properties and behavioural features of the model's components have been analysed and described. The model relatively accurately implements certain observed biological traits but idealises or fails to consider others. To simulate more rigorously the behaviour of animals in their natural habitat, model extensions considering for instance environment particularities or noise robustness should be considered.

Appendices

A Transfer Function

figures/0_1.png

Figure 9: Effect of α, β on the transfer function $g(h|\alpha, \beta)$. Assuming $\alpha \geq 0$. For $\alpha < 0$, the effect is the same but mirrored along the $h = 0$ axis.

B Unconnected neurons firing activity

figures/0_2.png

Figure 10: Temporal dynamics of unconnected neurons receiving the slowly oscillating input $I_i(t) = I_0 \sin(\omega t)$ over $T = 1000$ms. The plots show the mean number of spikes per 1 ms bin across the N neurons for $N = 100$, $N = 1000$ as well the the theoretical firing rate, from left to right. The input $I_i(t)$ is shown as a dashed line in each of the panels. All parameters are as the defaults suggested in the exercise.

C Ring and Line models

figures/1_6.png

Figure 11: Simulation with ring connectivity versus line connectivity (Gaussian turning curve). The parameters used for the Gaussian turning curve are $J_0 = -1$, $J_1 = 3$ and $\sigma = 1$.

D Mean bump location linear in I_0

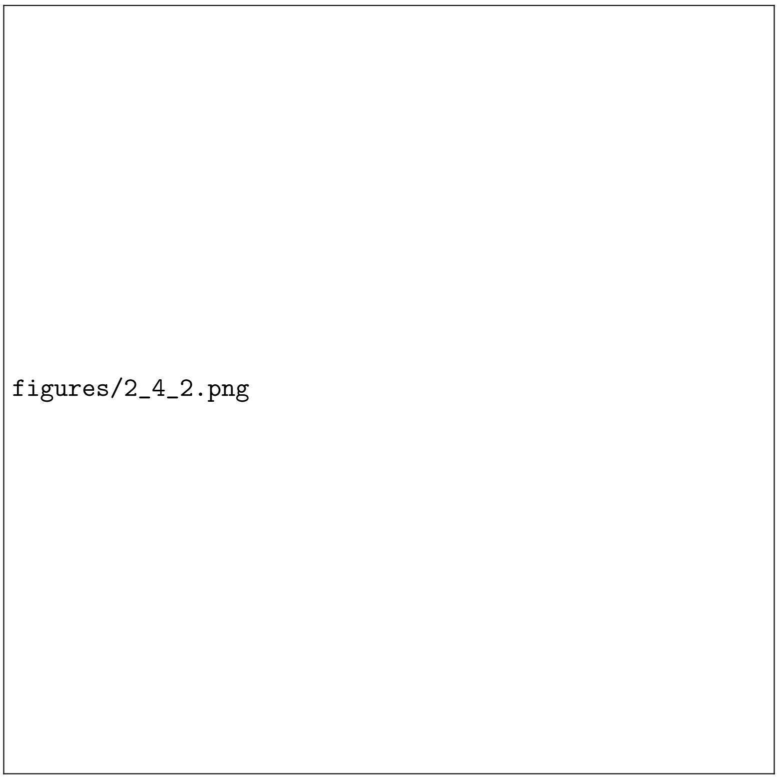

Figure 12: Relationship Between Input Strength I_0 and Final Bump Location $\theta_{\text{bump mean}}$.

E Finding value of $\mathbf{J_{head}}$

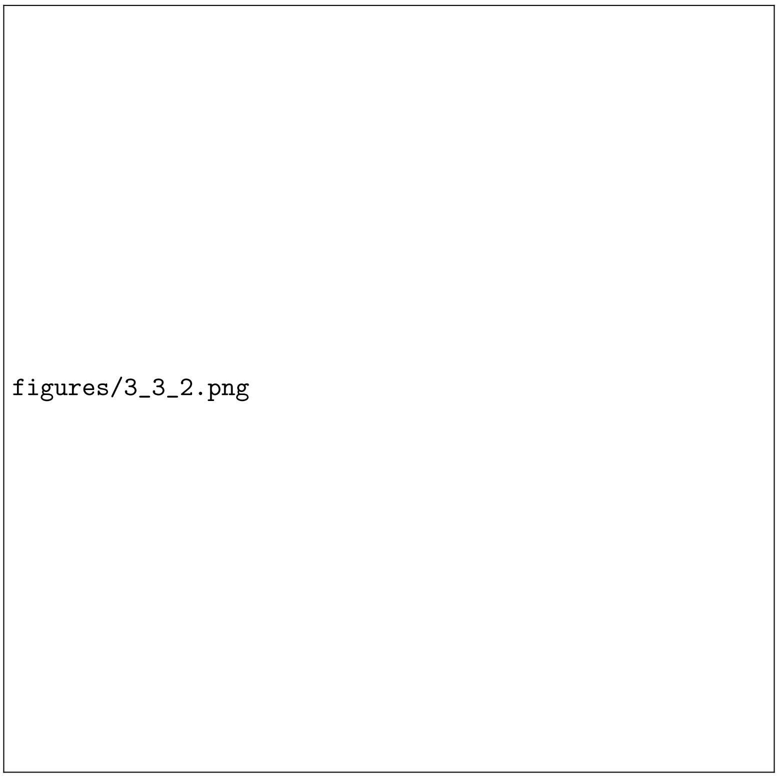

Figure 13: Determining the Optimal $\mathrm{J_{head}}$ value for Linear Regime Input from Head Direction Cells. The range of $\mathrm{J_{head}}$ tested spans from 0 to 7 pC.

F Simulation of a larger network

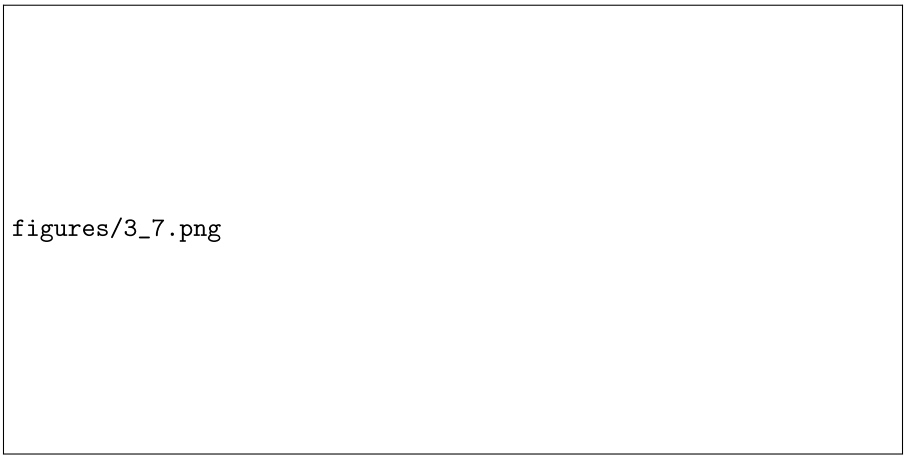

Figure 14: Comparing the capabilities of models with different number of neurons N to accurately decode a longer path (T $\geq$ 3s). The parameters of the models θ (15°) and τ (12 ms) were tuned by comparing the mean over 10 runs, of the sum of the Euclidean distances between the decoded and initial paths at each time-step. For θ, values of 5°, 10°, and 15° were tested, while τ was varied from 8 to 12.

References

[1] Mark Goldman, Albert Compte, and Xiao-Jing Wang. NEW ENCYCLOPEDIA OF NEUROSCIENCE.

[2] Thomas Haferlach, Jan Wessnitzer, Michael Mangan, and Barbara Webb. Evolving a Neural Model of Insect Path Integration. *Adaptive Behavior*, 15(3):273–287, September 2007.

[3] Ioannis Pisokas, Nina Kudryashova, and Matthias H Hennig. Can the Insect Path Integration Memory be a Bump Attractor?

[4] Raymond Wang and Louis Kang. Multiple bumps can enhance robustness to noise in continuous attractor networks. *PLOS Computational Biology*, 18:1–38, 10 2022.